AMENDMENTS TO THE CLAIMS

Please amend the claims as follows.

1. (Currently Amended) A method for accuracy-aware analysis of a program, comprising:

obtaining source code for the program comprising a floating-point variable;

instrumenting the source code to associate an accuracy-aware tracking structure with the floating-point variable to obtain instrumented source code with functionality to call a runtime logging utility;

compiling the instrumented source code to obtain instrumented compiled code;

executing the instrumented compiled code, wherein executing the instrumented compiled code comprises executing one of a plurality of operations on the floating-point variable to obtain a resultant value for the floating-point variable; and

executing the runtime logging utility to populate the accuracy-aware tracking structure, wherein for each operation performed on the floating-point variable, the runtime logging utility is configured to:

increment an operations variable in the accuracy-aware tracking structure corresponding to the operation performed on the floating-point variable,

determine a scaled mantissa for the resultant value, wherein the scaled mantissa representation for a floating-point variable corresponds to the integer represented by the first non-zero digit in the mantissa to the last non-zero-digit in the mantissa,

compare the scaled mantissa with the resultant value to determine whether the resultant value is exact,

quantify an error associated with the resultant value when the resultant value is not exact to obtain the error value associated with the resultant value, wherein the error value is a half unit in last place (HULP) value associated with the floating-point variable, wherein the HULP value is a base of a floating-point representation raised to a power of the number of bits causing the error in the floating-point variable, [[.]]

store the resultant value, the scaled mantissa, and the error value when the resultant value is not exact, and

store the resultant value and the scaled mantissa when the resultant value is exact.

2. (Original) The method of claim 1, further comprising:
   generating an accuracy-aware analysis report using the accuracy-aware tracking structure.

3. (Previously Presented) The method of claim 2, wherein the accuracy-aware analysis report includes at least one selected from the group consisting of the error value, the scaled mantissa, and the resultant value.

4. (Cancelled)

5. (Previously Presented) The method of claim 1, wherein the HULP value is determined using information obtained during renormalization.

6. (Previously Presented) The method of claim 3, wherein the error value comprises an upper limit interval value or a lower limit interval value.

7. (Previously Presented) The method of claim 1, wherein the operations variable comprises at least one selected from the group consisting of a multiplication variable, a division variable, and a square root variable.

8. (Previously Presented) The method of claim 1, wherein the accuracy-aware tracking structure further comprises a renormalization variable which track the number of addition and subtraction operations performed on the floating-point variable that do not involve left digit destruction.

9. (Cancelled)

10. (Previously Presented) The method of claim 1, further comprising:
    updating the error value using data obtained from quantifying the error associated with the resultant value, if the resultant value is not exact.

11. (Previously Presented) The method of claim 1, further comprising:
    determining whether the resultant value exceeds an accuracy threshold if the resultant value is not exact.

12. (Original) The method of claim 11, wherein execution of the compiled instrumented code halts if the accuracy threshold hold is exceeded.

13. (Original) The method of claim 11, wherein the accuracy threshold comprises at least one selected from the group consisting of a relative error threshold, an absolute error threshold, and a comparison test.

14. (Original) The method of claim 1, further comprising:
    setting an accuracy threshold for the program.

15. (Previously Presented) The method of claim 1, wherein instrumenting the source code comprises:
    parsing the source code to obtain the floating-point variable; and
    inserting additional source code to call the runtime logging utility.

16. (Cancelled)

17. (Original) The method of claim 1, wherein the floating-point variable is double type.

18. — 29. (Cancelled)

30. (Currently Amended) A computer system for performing accuracy-aware analysis on a program, comprising:

a processor;

a memory;

a storage device; and

software instructions stored in the memory for enabling the computer system under control of the processor, to:

obtain source code for the program comprising a floating-point variable;

instrument the source code to associate an accuracy-aware tracking structure with the floating-point variable to obtain instrumented source code with functionality to call a runtime logging utility;

compile the instrumented source code to obtain instrumented compiled code;

execute the instrumented compiled code, wherein executing the instrumented compiled code comprises executing one of a plurality of operations on the floating-point variable to obtain a resultant value for the floating-point variable; and

execute the runtime logging utility to populate the accuracy-aware tracking structure wherein for each operation performed on the floating-point variable, the runtime logging utility is configured to:

increment an operations variable in the accuracy-aware tracking structure corresponding to the operation performed on the floating-point variable,

determine a scaled mantissa for the resultant value, wherein the scaled mantissa representation for a floating-point variable corresponds to the integer by the first non-zero digit in the mantissa to the last non-zero-digit in the mantissa,

compare the scaled mantissa with the resultant value to determine whether the resultant value is exact,

quantify an error associated with the resultant value when the resultant value is not exact to obtain the error value associated with the resultant value, wherein the error value is a half unit in last place (HULP) value associated with the floating-point variable, wherein the HULP value is a base of a

floating-point representation raised to a power of the number of bits causing the error in the floating-point variable, [[.]]

store the resultant value, the scaled mantissa, and the error value when the resultant value is not exact, and

store the resultant value and the scaled mantissa when the resultant value is exact.

31. (Previously Presented) The computer system of claim 30, further comprising software instructions to:

    generate an accuracy-aware analysis report using the accuracy-aware tracking structure.

32. (Previously Presented) The computer system of claim 31, wherein the accuracy-aware analysis report includes at least one selected from the group consisting of the error value, the scaled mantissa, and the resultant value.

33. (Cancelled)

34. (Previously Presented) The computer system of claim 30, wherein the HULP value is determined using information obtained during renormalization.

35. (Previously Presented) The computer system of claim 30, wherein the operations variable comprises at least one selected from the group consisting of a multiplication variable, a division variable, and a square root variable.

36. (Cancelled)

37. (Previously Presented) The computer system of claim 30, wherein software instructions for instrumenting the source code comprises software instructions to:

    parse the source code to obtain the floating-point variable; and

    insert additional source code to call the runtime logging utility.